

## CLAIMS

- 1 1. A system for takeover of a Transport Control Protocol (TCP) connection by a  
2 second server from a first server comprising:  
3 a structure for the first and second servers adapted to access shared state infor-  
4 mation with respect to the connection;  
5 a device for comparing a data packet sequence number of an acknowledgement  
6 byte received by the second server with a sequence number related to the shared state in-  
7 formation; and  
8 a device for recreating the connection within the second server based upon the  
9 compared sequence numbers.
- 1 2. The system as set forth in claim 1 wherein the device for recreating includes an  
2 application program interface (API) for communicating with a plurality of protocols in  
3 the second server and providing a ready signal in response to a successful comparison by  
4 the device for comparing.
- 1 3. The system as set forth in claim 1 further comprising a connection checkpoint ap-  
2 plication program interface (API) for communicating with each of a plurality of protocols  
3 in the first server and for causing each of the plurality of protocols to append relevant  
4 state information to a data block passed to each of the plurality of protocols with respect  
5 to the connection so as to provide the relevant state information to the shared state infor-  
6 mation.
- 1 4. The system as set forth in claim 3 wherein the connection checkpoint API is  
2 adapted to bundle connection information with respect to a protocol having a plurality of  
3 related connections, the related connections involving both TCP protocol and User Da-  
4 tagram Protocol (UDP).

1 5. The system as set forth in claim 4 wherein the protocol having a plurality of re-  
2 lated connections comprises a protocol having a control connection and a data connec-  
3 tion.

1 6. The system as set forth in claim 5 wherein one or more of the data connections  
2 are carried over UDP or another non-TCP transport protocol.

1 7. The system as set forth in claim 6 wherein data packets on one or more of the data  
2 connections are adapted to be transmitted to one or more IP-multicast groups.

1 8. The system as set forth in claim 1 wherein the shared state information includes  
2 an identifier of the first server, the time at which the state information is gathered, a  
3 source IP address, a source TCP port, a destination IP address, a destination TCP port, an  
4 application layer protocol with respect to the connection, an initial packet sequence num-  
5 ber for the source, an initial packet sequence number for the destination, a current packet  
6 sequence number for the source, a current packet sequence number for the destination,  
7 and application layer information including the TCP sequence number for the first byte of  
8 an object and an identifier for the object.

1 9. A method for takeover of a Transport Control Protocol (TCP) connection by a  
2 second server from a first server comprising:  
3 generating shared state information with respect to the connection for access by  
4 the first server and the second server;  
5 comparing a received data byte sequence number from an acknowledgement byte  
6 received by the second server with a sequence number related to the shared state infor-  
7 mation; and  
8 recreating the connection within the second server based upon the compared se-  
9 quence numbers.

1 10. The method as set forth in claim 9 wherein recreating the connection includes  
2 communicating with a plurality of protocols in the second server to provide a ready signal  
3 in response to a successful comparison of the sequence numbers.

1 11. The method as set forth in claim 9 further comprising performing a connection  
2 checkpoint with an application program interface (API) so as to communicate with each  
3 of a plurality of protocols in the first server and so as to cause each of the plurality of  
4 protocols to append relevant state information to a data block passed to each of the plu-  
5 rality of protocols with respect to the connection, and to thereby provide the relevant state  
6 information to the shared state information.

1 12. The method as set forth in claim 11 further comprising relating, by the API, the  
2 compared sequence numbers in conjunction with the ready signal to a byte sequence  
3 number in an object referenced in the shared state information.

1 13. The method as set forth in claim 12 in which the relating includes resuming  
2 sending the data associated with an object over the restarted TCP connection.

1 14. The method as set forth claim 11 further comprising a network protocol stack in-  
2 cluding the plurality of protocols and checkpoint information based upon the connection  
3 checkpoint, and using an application program interface (API) on the second server to no-  
4 tify each of the protocols in the network stack to use the checkpoint information to  
5 thereby create an "unready" connection.

1 15. The method as set forth in claim 11 wherein the API is adapted to bundle connec-  
2 tion information with respect to a protocol having a plurality of related connections.

1 16. The method as set forth in claim 15 wherein the protocol having a plurality of re-  
2 lated connections comprises a protocol having a control connection and a data connec-  
3 tion.

1 17. The method as set forth in claim 16 wherein one or more of the data connections  
2 are carried over UDP or another non-TCP transport protocol.

1 18. The method as set forth in claim 17 wherein data packets on one or more of the  
2 data connections are adapted to be transmitted to one or more IP-multicast groups

1 19. The method as set forth in claim 9 wherein the shared state information includes  
2 an identifier of the first server, a time at which the shared state information is received, a  
3 source IP address, a source TCP port, a destination IP address, a destination TCP port, an  
4 application layer protocol with respect to the connection, an initial packet sequence num-  
5 ber for the source, an initial packet sequence number for the destination, a current packet  
6 sequence number for the source, a current packet sequence number for the destination,  
7 and application layer information.

1 20. The method as set forth in claim 9 wherein the recreating includes assignment of  
2 the second server to takeover the connection based upon a detecting a failure or overbur-  
3 dening of the first server.

1 21. Functional data embodied in one or more computer-readable media for takeover  
2 of a Transport Control Protocol (TCP) connection by a second server from a first server,  
3 the functional data comprising:

4 a device for generating shared state information with respect to the connection  
5 for access by the first server and the second server;

6 a device for comparing a data packet sequence number of an acknowledgement  
7 byte received by the second server with a sequence number related to the shared state in-  
8 formation; and

9 a device for recreating the connection within the second server based upon the  
10 compared sequence numbers.

1 22. In a networking protocol stack including a Transport Control Protocol (TCP)  
2 having associated TCP send sequence numbers:

3           an application interface to the networking stack adapted to allow an application to  
4   associate a byte of data to be sent on a TCP connection with one of either a current or  
5   future TCP send sequence number.

2025-01-23 11:46 AM